공학석사학위논문

경첩문을 여는 비행 매니퓰레이터에 대한 모델 예측 제어

Opening a Hinged Door with an Aerial Manipulator using Model Predictive Control

2020 년 8 월

서울대학교 대학원

기계항공공학부

이 동 재

Opening a Hinged Door with an Aerial Manipulator using Model Predictive Control

A Thesis

by

DONGJAE LEE

Presented to the Faculty of the Graduate School of
Seoul National University
in Partial Fulfillment
of the Requirements
for the Degree of

MASTER OF SCIENCE

Department of Mechanical & Aerospace Engineering

Seoul National University

Supervisor : Professor H. Jin Kim

August 2020

to my

FAMILY

with love

Abstract

Opening a Hinged Door with an Aerial Manipulator using Model Predictive Control

Dongjae Lee
Department of Mechanical & Aerospace Engineering
The Graduate School
Seoul National University

From aerial pick-and-place to aerial transportation, aerial manipulation has been extensively studied in recent years thanks to its mobility and dexterity, each of which is a merit of an aerial vehicle and a robotic arm. However, to fully harness the concept of aerial manipulation, more complex tasks including interaction with movable structures should also be considered. Among various types of movable structures, this paper presents a multirotor-based aerial manipulator opening a daily-life moving structure, a hinged door. Two additional issues that would arise in interaction with a movable structure are addressed: 1) a constrained motion of the structure, and 2) collision avoidance with a moving structure. To handle these issues, we formulate a model predictive control (MPC) problem with a system dynamics constraint and state constraints for collision avoidance. A coupled system dynamics of a multirotor-based aerial manipulator and a hinged door is derived and later simplified for faster computation in MPC. State constraints for collision avoidance with itself, a door, and a doorframe are generated. By implementing a constrained version of differential dynamic programming (DDP), we can generate reference trajectories through MPC in real-time. The proposed method is validated through simulation results and experiments with a real-like hinged door in which a disturbance observer (DOB) based robust motion controller is employed.

Keyword : Aerial manipulator, Model predictive control, Collision avoidance

Student Number : 2018-22356

Table of Contents

List of Figures

List of Tables

1 Introduction

Combining advantages of an aerial vehicle that can reach any position in three-dimensional space with advantages of a robotic arm that can perform physical interaction with an external environment, aerial manipulation has been studied extensively in recent years. [1] Various real-life applications were investigated with an unmanned aerial manipulator (UAM), including tool handling [2] and autonomous sensor installation [3]. Existing papers [2, 4, 5] consider interaction with a static structure; however, there exist few literature about movable structure interaction with an aerial manipulator which is one of inevitable tasks in unstructured, complex environment. If we can facilitate a UAM to move any movable surrounding structure, a more exhaustive exploration can be performed by accessing once unreachable places, pushing or pulling a movable structure; also, a more active response to disaster or rescue operation can be achieved with its augmented versatility.

At least two additional issues should be addressed for interacting with a movable structure compared to an interaction with a static counterpart: 1) structure dynamics, and 2) collision avoidance. In contrast to a static structure, a movable structure contains its own dynamics. Proper modeling or estimation on the movement of the structure is required for both guiding the structure to the desired position and avoiding collisions with this dynamic structure.

As part of such topics, this paper handles the problem of a UAM opening a hinged door. It

deals with a particular application of a multirotor-based UAM operating a movable structure. This problem requires additional consideration on the movement of the structure which would involve intrinsic constraints, which is, in our case, a hinge constraint. Moreover, crash prevention with a dynamic structure, the door, and a static structure, a doorframe, is considered along with a self-collision avoidance in generating a collision-free trajectory. After modeling the combined dynamics of the UAM and the hinged door, we applied model predictive control (MPC) to ensure dynamic feasibility and collision avoidance of a generated trajectory. State constraints which are formulated from kinematic relationships are imposed on the optimal control problem, and the generated trajectory from MPC is tracked by a disturbance observer-based robust controller designed in [6].

This thesis integrates the author's previous works [7] and [8] which are each published in *2019 International Conference on Control, Automation and Systems (ICCAS)* and to be published in *2020 International Conference on Robotics and Automation (ICRA)*. Much of the materials are based on the previous works, but a further extension has also been made in that more detailed explanations and additional simulation results for various door parameters are presented.

1.1 Literature review

In [2], tool handling aerial manipulator is suggested while in contact with a vertical surface. For a contact-based inspection task, a motion planning algorithm presented in [9] is validated with experiments using a multidirectional thrust aerial vehicle [10]. An omnidirectional aerial vehicle for contact-based inspection on a curved surface is presented in [5] with state estimation and disturbance rejection. However, none of these considers a moving structure, and since the planner suggested in [9] is based on a sampling-based method, it cannot be applied for an online scenario.

There exist some studies about an aerial manipulator coping with a movable structure. In [11], a dual-armed aerial manipulator is employed to perform three different tasks, and the one about a movable structure is a valve-turning task. However, in this scenario, only a rotational motion in the z-axis is required for the task, and certainly, more issues have to be considered for less constrained tasks. In [12], the author presents an experiment of an aerial manipulator operating an unknown drawer. Coupled dynamics between the UAM and the drawer are derived, and the

desired force is computed through the velocity of the drawer. However, a structure with only a translational motion is considered, and collision avoidance is not explicitly addressed. In [13], an aerial vehicle opens a hinged door with a proposed mechanism. Though suitable for this particular purpose, versatility that a general UAM contains seems to lack in this approach.

MPC has been utilized not only as an optimal controller for aerial vehicles but also as an optimal planner. In [14], an explicit MPC is used to generate offline control policies for an aerial manipulator interacting with a static structure. Later, in [15], MPC with SLQ is presented for an online optimal control input computation. Similar techniques are applied in many other papers to produce an online optimal trajectory for a multirotor with a suspended load [16], and for a multirotor with a network delay [17]. In this paper, MPC is utilized as a sub-optimal planner generating a trajectory complying with dynamics and constraints.

1.2 Thesis contribution

This paper deals with the problem of opening a hinged door with an aerial manipulator while avoiding collisions which, to the best of the author's knowledge, has not been considered in the existing literature. Simplified dynamics and state constraints are formulated to construct an MPC problem, and by adopting a constrained version of differential dynamic programming, dynamically feasible and online-applicable trajectories satisfying constraints are generated. Finally, our proposed approach is validated through simulations and real experiments.

1.3 Thesis outline

The remainder of this paper is structured as follows. In Section 2, equations of motion of the whole system including kinematics, dynamics ,and simplified dynamics are derived. Section 3 starts with an MPC problem formulation, followed by introduction to three types of state constraints and an optimal control solver. Control framework for the whole system is introduced in Section 4. Experimental setup and results are explained in Section 5, while conclusions are drawn in Section 6.

2

Equations of motion

In this chapter, based on previous works [7, 8], equations of motion (EoM) of the whole system including the multirotor-based UAM and the hinged door are derived. A general dynamical model of the system is first introduced by Lagrange equation, and a simplified dynamics is then suggested for real application.

2.1 Assumption

Following assumptions are made when deriving the EoM:

- Servomotors in the robotic arm have negligible dynamics.
- The end-effector's tip and the door's surface are rigidly connected.
- Physical properties of the door (i.e., mass moment of inertia, width, height, and position) are known.

Although the first assumption may break with an aggressive motion of the robotic arm, we can assure the assumption through applying a DOB-based robust controller which alleviates effects from the motion of the robotic arm to the multirotor. Note that the second and third assumptions may be relieved by introducing a force control approach which can directly handle the interaction

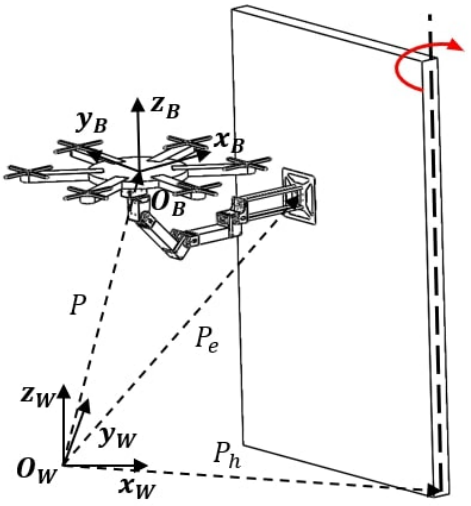

Figure 2.1: Reference frames of the aerial manipulator in the door opening scenario

between the aerial manipulator and the moving structure or a parameter estimation algorithm for estimating unknown physical parameters, but for the time being, we limit the scope of this paper through these two assumptions. With the followed assumptions, equations of motion and constraints for the system are developed.

2.2 Kinematics

To describe states and parameters of the system, as in Figure 2.1, we first define coordinate frames $\mathscr{F}_W = \{O_W, x_W, y_W, z_W\}$ and $\mathscr{F}_B = \{O_B, x_B, y_B, z_B\}$ which are a world fixed frame and a body-fixed frame, respectively. With a rotation matrix R_t from the frame $\mathscr{F}_B$ to $\mathscr{F}_W$ and a position vector $\mathbf{d}$ of the end-effector from O_B described in $\mathscr{F}_B$, the kinematic constraint between the UAM and the door can be written as the following:

$$\mathbf{P} + R_t\mathbf{d} = \mathbf{P_h} + [D_V \cos\alpha \; D_V \sin\alpha \; D_H]^T \tag{2.1}$$

where $\mathbf{P}$, $\mathbf{P_h}$, D_V, D_H and α are position of the center of mass (COM) of the aerial manipulator in $\mathscr{F}_W$, position of the door hinge in $\mathscr{F}_W$, the shortest distance between the end-effector and the door hinge, the height of the end-effector in $\mathscr{F}_W$, and the door angle.

Revisiting the first assumption in section 2.1, dynamics of the robotic arm can be neglected while deriving the whole system dynamics. Therefore, we draw the system dynamics in a decoupled

structure: dynamics of the whole system without the robotic arm and kinematics of the robotic arm. The first portion of the system dynamics is defined with a partial configuration of the whole system $\mathbf{q} = [\mathbf{\Phi}\ \alpha]^T \in \mathbb{R}^4$, where $\mathbf{\Phi}$ is an Euler angle of the frame $\mathscr{F}_B$. The rest are described with the angle of each joint in the robotic arm, which is denoted as $\mathbf{H} = [\eta_1\ \eta_2\ \eta_3\ \eta_4]^T \in \mathbb{R}^4$.

The first portion of the system dynamics is drawn with an Euler-Lagrange equation. To construct a mass matrix in Euler-Lagrange equation, Jacobian matrices mapping $\dot{\mathbf{q}}$ to $\dot{\mathbf{P}}$, $\mathbf{\Omega}$, and $\dot{\alpha}$ are defined as J_t, J_r, and J_α where $\dot{\mathbf{P}} = J_t\dot{\mathbf{q}} - R_t\dot{\mathbf{d}}$, $\mathbf{\Omega} = J_r\dot{\mathbf{q}}$, and $\dot{\alpha} = J_\alpha\dot{\mathbf{q}}$. $\dot{*}$ denotes a time derivative of $*$, and $\mathbf{\Omega}$ is a body angular velocity of $\mathscr{F}_B$. The Jacobian matrices can be analytically calculated as follows:

$$\begin{aligned} J_t &= \begin{bmatrix} & \begin{bmatrix} -D_V \sin\alpha \\ D_V \cos\alpha \\ 0 \end{bmatrix} \\ R_t\hat{\mathbf{d}}Q & \end{bmatrix} \in \mathbb{R}^{3\times 4} \\ J_r &= \begin{bmatrix} Q\ 0_{3\times 1} \end{bmatrix} \in \mathbb{R}^{3\times 4} \\ J_\alpha &= \begin{bmatrix} 0_{1\times 3}\ 1 \end{bmatrix} \in \mathbb{R}^{1\times 4} \end{aligned} \tag{2.2}$$

where $\hat{*}$ is a hat operator satisfying $a \times b = \hat{a}b$ for any vectors $a,\ b \in \mathbb{R}^3$, and Q is the mapping matrix from Euler angle's time derivative $\dot{\mathbf{\Phi}}$ to the body angular velocity $\mathbf{\Omega}$.

Since the second portion of the system dynamics only considers kinematics of the robotic arm, defining a virtual control input of the robotic arm as $\dot{\mathbf{H}}_d = [\dot{\eta}_{1,d}\ \dot{\eta}_{2,d}\ \dot{\eta}_{3,d}\ \dot{\eta}_{4,d}]^T \in \mathbb{R}^4$, we modeled the motion of the robotic arm as a single integrator model as follows:

$$\dot{\mathbf{H}} = \dot{\mathbf{H}}_d. \tag{2.3}$$

This simple model can be reasonably assumed through applying a velocity control scheme with a high P gain to each servomotor of the robotic arm.

2.3 Dynamics

In this section, the first portion of the system dynamics is fully derived and is integrated with the equation (2.3) to formulate the whole system dynamics.

To apply Lagrange equation with respect to $\mathbf{q}$ and $\dot{\mathbf{q}}$, kinetic and potential energies are calculated as follows:

$$\begin{aligned}\mathcal{K} &= \frac{1}{2}\left(\dot{\mathbf{P}}^T M_A \dot{\mathbf{P}} + \boldsymbol{\Omega}^T I_A \boldsymbol{\Omega} + \dot{\alpha}^T I_D \dot{\alpha}\right) \\ \mathcal{P} &= m_A g \mathbf{P}^T \mathbf{e}_3 + \frac{1}{2}\left(\alpha^T K_D \alpha\right)\end{aligned} \tag{2.4}$$

where $M_A = diag(m_A, m_A, m_A)$, and $\mathbf{e}_3 = [0\ 0\ 1]^T$. m_A is the mass of the aerial manipulator, and K_D is a spring constant of the door. By calculating the Lagrange equation with external forces, the EoM of the multirotor and the door can be drawn analytically from the below equation:

$$\frac{d}{dt}\frac{\partial \mathcal{L}}{\partial \dot{\mathbf{q}}} - \frac{\partial \mathcal{L}}{\partial \mathbf{q}} = \tau_{\mathbf{q}} + \tau_{\mathbf{ext}}, \quad \mathcal{L} = \mathcal{K} - \mathcal{P} \tag{2.5}$$

By solving the equation (2.5) with the equations (2.2) and (2.4), the dynamics can be arranged as the following:

$$M_q \ddot{\mathbf{q}} + C_q \dot{\mathbf{q}} + G_q = \tau_q + \tau_{ext}, \tag{2.6}$$

where

$$\begin{aligned}\tau_q &= \begin{bmatrix} Q^T & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \begin{bmatrix} -d_y \\ d_x \\ 0 \end{bmatrix} I_{3\times 3} \\ -D_V R_0 \quad 0_{1\times 3} \end{bmatrix} \begin{bmatrix} f_t \\ \tau_x \\ \tau_y \\ \tau_z \end{bmatrix}, \\ R_0 &= R_1 \sin\alpha + R_2 \cos\alpha, \\ R_1 &= \sin\phi \sin\psi + \cos\phi \sin\theta \cos\psi, \\ R_2 &= \sin\phi \cos\psi + \cos\phi \sin\theta \sin\psi.\end{aligned} \tag{2.7}$$

Here M_q, C_q, and G_q are mass, Coriolis, and gravitational matrices. d_x and d_y are the first and second component of $\mathbf{d}$, and f, τ_x, τ_y, τ_z are thrust, moment in x_B-axis, moment in y_B-axis, and moment in z_B-axis from the multirotor, respectively. ϕ, θ, and ψ are roll, pitch and yaw Euler angle of the frame $\mathscr{F}_B$. τ_{ext} denotes an external force and torques such as friction acting on the door which can be modeled as $\tau_{ext} = [0\ 0\ 0\ \ -D_D\dot{\alpha}]^T$ with a damping constant D_D.

Combining equations (2.6) and (2.3), the entire system's EoM can finally be drawn with state

$\mathbf{x} = [\mathbf{q}\ \dot{\mathbf{q}}\ \mathbf{H}]^T \in \mathbb{R}^{12}$ and control input $\mathbf{u} = [f_t\ \tau_x\ \tau_y\ \tau_z\ \dot{\mathbf{H}}_d]^T \in \mathbb{R}^8$ as follows:

$$\dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u}) = \begin{bmatrix} \dot{\mathbf{q}} \\ M_q^{-1}(-(C_q\dot{\mathbf{q}} + G_q) + \tau_q + \tau_{ext}) \\ \dot{\mathbf{H}}_d \end{bmatrix} \tag{2.8}$$

2.4 Simplified dynamics

A simplified dynamical model for the whole system is derived in this section. Necessity for the simplified dynamics can be summarized as follows. Since our approach to handle the door opening scenario with a UAM includes an optimal planner in a receding horizon manner, a more concise model though including all sufficient information of the system dynamics would be more advantageous in terms of computation time. With some additional assumptions, the number of system states can be reduced, and numerical matrix inversion is no longer required which is mandatory in the equation (2.8).

The simplified version is derived with the following two assumptions:

- Multirotor's rotational dynamics is negligible.
- A UAM is in a quasi-static state in translational motion during its operation.

The first assumption can be justified from the fact that an onboard controller is able to control the angular velocity of the vehicle sufficiently fast thaks to multirotor's low rotational inertia and its ability to generate high torque. Several other papers [18, 19] include this assumption in their dynamical model for multirotor planning. Although our system model differs from that of nominal multirotor dynamics, by virtue of the robust controller, nominal dynamics of the multirotor can be recovered [6], through which the assumption can now be validated. Secondly, considering the door's relatively higher inertia compared to that of the UAM, the second assumption can be assumed without losing generality. Similar research carried out for operating a drawer with a UAM in [12] also assumed this quasi-static motion to generate the desired path.

Thanks to the first assumption, we can redefine the states and inputs for the system to $\mathbf{x_s} = [\mathbf{\Phi}\ \alpha\ \dot{\alpha}\ \mathbf{H}]^T \in \mathbb{R}^9$, and $\mathbf{u_s} = [f_t\ \mathbf{\Omega}_d\ \dot{\mathbf{H}}_d]^T \in \mathbb{R}^8$, where $\mathbf{\Omega}_d$ denotes a desired angular velocity of the UAM in $\mathscr{F}_B$. Furthermore, according to the second assumption, introducing an

action-reaction force F_R between the UAM and the hinged door, F_R can be written as $\mathbf{F}_R = -m_A g\mathbf{e}_3 + f_t R_t \mathbf{e}_3$ where g is a gravitational acceleration. Then, the nominal dynamics of the hinged door can be represented as $I_D\ddot{\alpha} + D_D\dot{\alpha} + K_D\alpha = -\mathbf{n}_D{}^T\mathbf{F}_R D_V$ where and $\mathbf{n}_D = [\sin\alpha \ -\cos\alpha \ 0]^T$ stands for a unit normal vector to the door surface pointing to the opposite direction to the UAM. Therefore, the door's angular acceleration can be rearranged as $\ddot{\alpha} = g(\mathbf{\Phi}, \alpha, f_t) \triangleq 1/I_D\left(-(D_D\dot{\alpha} + K_D\alpha) - f_t D_V \mathbf{n}_D^T R_t \mathbf{e}_3\right)$, and the whole system dynamics with the redefined states $\mathbf{x_s}$ and inputs $\mathbf{u_s}$ are as follows:

$$\dot{\mathbf{x}}_s = f_s(\mathbf{x}_s, \mathbf{u}_s) = \begin{bmatrix} W(\mathbf{\Phi})\mathbf{\Omega}_d \\ \dot{\alpha} \\ g(\mathbf{\Phi}, \alpha, f_t) \\ \dot{\mathbf{H}}_d \end{bmatrix} \tag{2.9}$$

where $W(\mathbf{\Phi})$ is a mapping matrix such that $\dot{\mathbf{\Phi}} = W(\mathbf{\Phi})\mathbf{\Omega}$.

3

Model predictive control

This chapter elaborates on MPC problem formulation. Starting with a general problem formulation, hard constraints for avoiding a collision, and an optimal control solver enabling a real-time application are presented sequentially. By repeatedly solving an optimal control problem for a particular time horizon, our MPC algorithm could generate a dynamically feasible and safe trajectory applicable to a real-world experiment.

3.1 Problem formulation

Given the current state $\mathbf{x}_0$ and initial input sequence $U_0 = \{\mathbf{u}_0, \cdots, \mathbf{u}_{N-1}\}$, an optimal control problem for a discrete time system can be formulated as follows:

$$\begin{aligned} \min_{U} \ & \mathrm{J}(\mathbf{x}_0, U) = l_f(\mathbf{x}_N) + \sum_{i=0}^{N-1} l(\mathbf{x}_i, \mathbf{u}_i)\Delta t, \\ & \text{s.t. } \mathbf{x}_{k+1} = f_D(\mathbf{x}_k, \mathbf{u}_k), \\ & \mathbf{x}_k \in \mathcal{X}_k \subset \mathbb{R}^{n_\mathbf{x}}, \\ & \mathbf{u}_k \in \mathcal{U}_k \subset \mathbb{R}^{n_\mathbf{u}} \end{aligned} \tag{3.1}$$

where Δt and N are a time interval and a time horizon. $f_D(\mathbf{x}_k, \mathbf{u}_k)$, $\mathcal{X}$, and $\mathcal{U}$ each denotes a discretized dynamics, a state constraint set, and an input constraint set with $n_\mathbf{x} = 9$ and $n_\mathbf{u} = 8$.

Dynamics f_D, states $\mathbf{x}_k$, and inputs $\mathbf{u}_k$ are discretized based on the simplified model in the equation (2.9).

3.1.1 Objective function

Since the door state is included in the system state $\mathbf{x}_k$, through the quadratic cost function with respect to state and input error, state and input trajectories for opening the door while spending minimum energy can be generated. The terminal cost l_f and the stage cost l are defined with each variable's desired value denoted with a superscript of d as follows:

$$\begin{aligned} l_f(\mathbf{x}_N) =& \frac{1}{2}\|\mathbf{x}_N - \mathbf{x}_N^d\|_L^2, \\ l(\mathbf{x}_i, \mathbf{u}_i) =& \frac{1}{2}\|\mathbf{x}_i - \mathbf{x}_i^d\|_Q^2 + \frac{1}{2}\|\mathbf{u}_i - \mathbf{u}_i^d\|_R^2 \end{aligned} \tag{3.2}$$

where $\|x\|_A^2 \triangleq x^T A x$.

3.1.2 Hard constraints

To ensure safe operation, hard constraints are generated. Collision avoidance constraints can be designed only with state variables, and through detailed derivation in the next section, they can be expressed as a set of inequalities $c_{state}(\mathbf{x}) \geq 0$. Thus, general expression for input and state constraint sets $\mathcal{U}_k$ and $\mathcal{X}_k$ can be simplified into $\mathcal{U}_k = \mathbb{R}^{n_u}$ and $\mathcal{X}_k = \{\mathbf{x} | c_{state}(\mathbf{x}_k) \geq 0\}$.

3.2 Collision avoidance constraints

To ensure a safe operation for the scenario, hard constraints for collision avoidance should be considered. Figure 3.1 shows three types of collision that could possibly occur during interaction with a hinged door. In this section, to avoid such situations, hard constraints for avoiding each type of collision in Figure 3.1 are handled: self-collision avoidance, door collision avoidance, and doorframe collision avoidance. These constraints are first derived using kinematic relationship and organized into inequality constraints $c_{state}(\mathbf{x}_k) \geq 0$ as discussed in subsection 3.1.2.

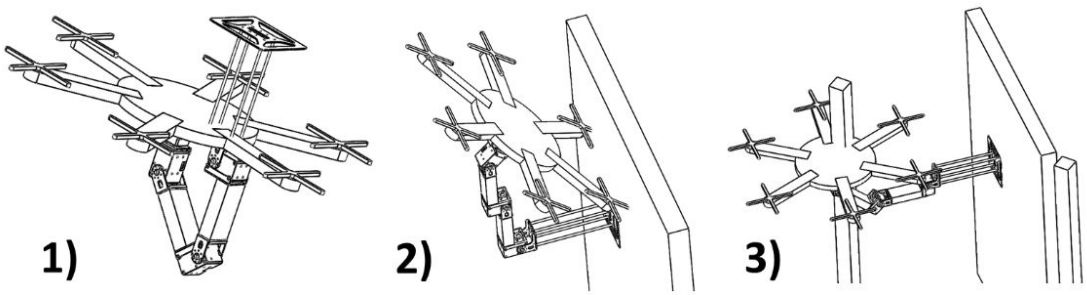

Figure 3.1: Three types of collision during interaction with a hinged door: 1) self-collision, 2) door collision, and 3) doorframe collision. In the first type of collision, the collision between the multirotor frame and the robotic arm is considered while in the second type of collision, the collision between the multirotor frame and the door is illustrated. Finally, in the thrid type, the collision between the aerial manipulator and the doorframe is depicted.

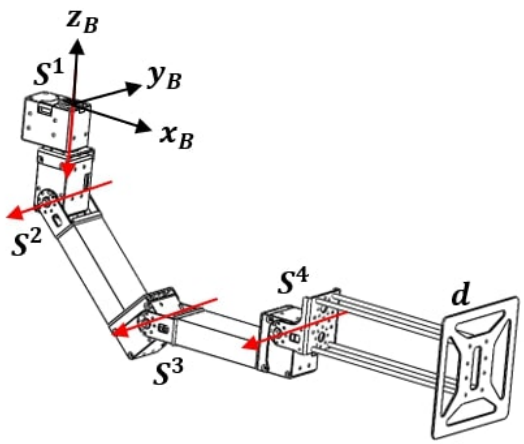

Figure 3.2: Configuration of the robotic arm. Positions of the joints are described as $\mathbf{S}^1$, $\mathbf{S}^2$, $\mathbf{S}^3$, and $\mathbf{S}^4$ in $\mathscr{F}_B$ while the rotation axes are depicted with the red arrows.

3.2.1 Self collision avoidance

First of all, unlike a general multirotor, our UAM's additional freedom in the robotic arm should be carefully managed to avoid a crash between the multirotor airframe and the robotic arm. To ensure this self-collision avoidance, following constraints are devised:

$$\begin{aligned} \mathbf{S}_z^3 &\leq 0 \\ \mathbf{S}_z^4 &\leq 0 \\ \mathbf{d}_z &\leq 0 \end{aligned} \tag{3.3}$$

where $\mathbf{S}^3$, $\mathbf{S}^4$, and $\mathbf{d}$ are position vectors of the 3^{rd} servomotor, the 4^{th} servomotor, and the end-effector of the robotic arm described in $\mathscr{F}_B$ while having their origins at O_B as in the Fig. 3.2, and the subscript z in $*_z$ denotes the third component of a vector $*$. Since O_B is assumed to be centered at the CoM of the multirotor, and these vectors are all described in $\mathscr{F}_B$, the above constraints imply that the robotic arm must always stay below the airframe of the multirotor. Note that all three position vectors $\mathbf{S}^3$, $\mathbf{S}^4$, and $\mathbf{d}$ are a function of $\mathbf{H}$ which can be derived with forward kinematics; therefore, these constraints can be formulated only with system state.

3.2.2 Door collision avoidance

The second constraint, which is avoiding collision with the door, is constructed as follows:

$$\mathbf{n}_D^T(R_t\mathbf{d}) \geq R_A \max_{\theta} \left\{ \mathbf{n}_D^T \left(R_t \begin{bmatrix} \cos\theta & \sin\theta & 0 \end{bmatrix}^T \right) \right\} \tag{3.4}$$

In the equation (3.4), the left-hand side indicates the shortest distance between the CoM of the UAM and the door surface, and the right-hand side quantifies the distance between the CoM of the UAM and the multirotor's airframe closest to the door surface. R_A is a radius of the multirotor frame including a propeller. If we introduce $\mathbf{n}_D^B \triangleq R_t^T\mathbf{n}_D \in \mathbb{R}^3$, a door surface unit normal vector in $\mathscr{F}_B$, the equation (3.4) can be arranged as follows:

$$(\mathbf{n}_D^B)^T\mathbf{d} \geq R_A\sqrt{(\mathbf{n}_{Dx}^B)^2 + (\mathbf{n}_{Dy}^B)^2} \tag{3.5}$$

Since all variables in this constraint are functions of state x_s only, this second constraint can also be formulated only with the system state.

3.2.3 Doorframe collision avoidance

The last doorframe avoiding constraint is formulated as the following:

$$\mathbf{P}_{Hy} + R_A \leq \mathbf{P}_y \leq \mathbf{P}_{Hy} + D_w - R_A \tag{3.6}$$

where subscript y of $*_y$ denotes the second component of a vector $*$. D_w is a length of the door width. This equation can be further organized using the equation (2.1) and can only be expressed with the state as

$$D_V \sin\alpha + R_A + D_w \leq \mathbf{d}_y^W \leq D_V \sin\alpha - R_A \tag{3.7}$$

where $\mathbf{d}^W \triangleq R_t\mathbf{d}$ is the end-effector position vector from the CoM of the UAM described in $\mathscr{F}_W$.

3.3 Optimal control solver

This section briefly introduces the optimal control solver, differential dynamic programming (DDP) with augmented Lagrangian method. Based on the problem and constraints formulated in the sections 3.1 and 3.2, an existing algorithm of the constrained version of DDP in [20] is employed to handle nonlinear dynamics with state and input constraints. This solver could generate an optimized nominal trajectory satisfying constraints in about 30 Hz with a time horizon of 1 second.

3.3.1 Differential dynamic programming

Optimal control problem with discretized nonlinear dynamical system $\mathbf{x}_{k+1} = f_D(\mathbf{x}_k, \mathbf{u}_k)$ and no constraints can be formulated as follows:

$$\begin{aligned} \min_U \ \mathrm{J}(\mathbf{x}_0, U) &= l_f(\mathbf{x}_N) + \sum_{i=0}^{N-1} l_i(\mathbf{x}_i, \mathbf{u}_i) \\ \text{s.t. } \mathbf{x}_{k+1} &= f_D(\mathbf{x}_k, \mathbf{u}_k) \end{aligned} \tag{3.8}$$

The goal is to find an optimal input sequence $U^* = \{\mathbf{u}_0^*, \cdots, \mathbf{u}_{N-1}^*\}$ that minimizes the objective function $\mathrm{J}(\mathbf{x}_0, U)$. Among numerous methods for solving this problem, one approach would be dynamic programming. Using Bellman's principle of optimality, *optimal cost-to-go function* $V_k(\mathbf{x}_k)$

can be defined as

$$\begin{aligned} V_N(\mathbf{x}_N) &= l_f(\mathbf{x}_N) \\ V_k(\mathbf{x}_k) &= \min_{\mathbf{u}_k} l_k(\mathbf{x}_k, \mathbf{u}_k) + V_{k+1}(f_D(\mathbf{x}_k, \mathbf{u}_k))) \end{aligned} \tag{3.9}$$

Through above *optimal cost-to-go function* $V_k(\mathbf{x}_k)$, one can find an optimal input sequence U^* by solving the minimization problem backwardly, starting from $V_N(\mathbf{x}_N)$. However, due to curse of dimensionality, naive implementation of this dynamic programming approach cannot solve the minimization problem in real time for high dimensional system.

To circumvent this problem, differential dynamic programming (DDP) has been suggested to find a suboptimal solution. Instead of finding a global minimum by considering all nonlinearities in the minimization argument in the equation (3.9), DDP locally approximates the argument using second order Taylor expansion, through which an analytic local minimum can be found. Since the local minimum is exactly the *optimal cost-to-go function* in the previous step, by solving the minimization problem recursively backward, a sub-optimal input sequence can be obtained, and this process is called backward pass. However, the input sequence generated from this one backward pass might result in poor performance due to approximation error; therefore, a process called forward pass is added to compose an inner loop together with the backward pass, and an outer loop iterates the inner loop until convergence. The detailed algorithm is explained in [21].

3.3.2 DDP with augmented Lagrangian method

The DDP algorithm in subsection 3.3.1 cannot be directly applied to our optimal control problem formulated in section 3.1 due to inequality hard constraints. To handle this hard constraints in DDP algorithm, we implemented an augmented Lagrangian method to the original DDP algorithm, proposed in [20]. This subsection presents the key concepts of the algorithm.

The optimal control problem in equation 3.1 can be rewritten as

$$\begin{aligned} \min_{U} \ \mathrm{J}(\mathbf{x}_0, U) &= l_f(\mathbf{x}_N) + \sum_{i=0}^{N-1} l_i(\mathbf{x}_i, \mathbf{u}_i)\Delta t \\ \text{s.t. } \mathbf{x}_{k+1} &= f_D(\mathbf{x}_k, \mathbf{u}_k) \\ c(\mathbf{x}_k, \mathbf{u}_k) &\geqq 0 \end{aligned} \tag{3.10}$$

where $c(\mathbf{x}_k, \mathbf{u}_k) \in \mathbb{R}^{n_c}$ is a vertical concatenation of all constraints, which is just $c_{state}(\mathbf{x}_k)$ in our case, and n_c is the total number of hard constraints.

Exploiting augmented Lagrange method, the constrained optimal control problem in the equation (3.10) can be reorganized as unconstrained optimal control problem with the system dynamics:

$$\min_{U} \ \mathcal{L}_a = \mathrm{J}(\mathbf{x}_0, U) + \sum_{i=0}^{N-1} \{c(\mathbf{x}_i, \mathbf{u}_i)^T I_{\mu_i} c(\mathbf{x}_i, \mathbf{u}_i) + \lambda_i^T c(\mathbf{x}_i, \mathbf{u}_i)\} \tag{3.11}$$

where $\lambda_i \in \mathbb{R}^{n_c}$ is a vector-form Lagrange multiplier at the i^{th} time step, and the matrix $I_{\mu_i} \in \mathbb{R}^{n_c \times n_c}$ is a diagonal matrix whose diagonal terms are defined as:

$$I_{\mu_i}(j,j) = \begin{cases} \mu_i^j & \text{if } c_j(\mathbf{x}_i, \mathbf{u}_i) < 0 \text{ or } \lambda_i^j > 0 \\ 0 & \text{otherwise} \end{cases} \tag{3.12}$$

With the above unconstrained optimization problem (3.11), nonlinear inequality constraints on state and input can be handled, and the original DDP algorithm derived in subsection 3.3.1 can be applied. The Lagrange multiplier λ is updated if any magnitude of constraint violation is larger than a threshold value, and otherwise update μ. μ is updated according to a predefined schedule while λ is updated using the first-order necessary condition for optimality as follows:

$$\lambda_i^j \leftarrow \lambda_i^j - \mu_i^j c_j(\mathbf{x}_k, \mathbf{u}_i) \tag{3.13}$$

The detailed algorithm can be found in [20].

4

Experimental setup

This chapter covers three additional issues that should be prepared before conducting real experiments: first, measuring or estimating the door state, second, controlling the multirotor, and third, hardware setup. The door state are estimated through kinematic relationship in the equation (2.1), and a disturbance observer(DOB)-based robust controller is applied to control the multirotor. Finally, a customized multirotor-based aerial manipulator with a 4 degree-of-freedom robotic arm is developed for experiments.

4.1 Door state estimation

To execute the MPC algorithm presented in the chapter 3, the door state α and $\dot{\alpha}$ are required. With measured state $\mathbf{P}$, $\dot{\mathbf{P}}$, $\mathbf{\Phi}$, $\mathbf{\Omega}$, revisiting the kinematic relationship in the equation (2.1), door state can be acquired as

$$\begin{aligned} \alpha &= \arctan\left(\frac{-\mathbf{P}_{hy}+\mathbf{P}_y+\mathbf{d}_y^W}{-\mathbf{P}_{hx}+\mathbf{P}_x+\mathbf{d}_x^W}\right) \\ \dot{\alpha} &= \begin{cases} \dfrac{\dot{\mathbf{P}}_x+\mathbf{d}_{Rx}^W+\dot{\mathbf{d}}_x^W}{-D_V\sin\alpha}, & \text{if } \alpha \neq n\pi \\ \dfrac{\dot{\mathbf{P}}_y+\mathbf{d}_{Ry}^W+\dot{\mathbf{d}}_y^W}{D_V\cos\alpha}, & \text{otherwise} \end{cases} \end{aligned} \tag{4.1}$$

where $n \in \mathbb{Z}$, $\dot{\mathbf{d}}^W = R_t\dot{\mathbf{d}}$, and $\mathbf{d}_R^W = R_t\hat{\Omega}\mathbf{d}$. This state conversion is based on the kinematic equation (2.1) and its time derivatives.

4.2 Multirotor robust controller

Since our task of opening a hinged door involves non-negligible interaction forces, a simple trajectory tracking controller may not be able to guarantee stability during operation. Therefore, we applied a robust controller with an ability to compensate for the estimated disturbance proposed in [6].

The controller consists of two parts: 1) extended high gain observer to recover nominal dynamics of the underactuated subsystem, and 2) disturbance observer (DOB) to recover nominal dynamics of the fully actuated subsystem. The underactuated sub-system is related to a translational motion of the whole system, whose configuration is denoted as $\mathbf{x}_c = [x_c \ y_c]^T \in \mathbb{R}^2$ where x_c and y_c are x and y positions of the COM of the aerial manipulator. The configuration of the rest fully actuated subsystem $\mathbf{z}$ is described with a position z of the COM of the multirotor and Euler angle $\mathbf{\Theta}$ of the multirotor, which in short, $\mathbf{z} = [z \ \mathbf{\Theta}]^T \in \mathbb{R}^4$.

Borrowing notations from the referred paper, nominal dynamics of the underactuated subsystem and the fully actuated subsystem are written as follows:

$$\begin{aligned} \ddot{\mathbf{x}}_c &= \bar{G}_x\mathbf{\Phi} \\ \ddot{\mathbf{z}} &= \bar{F}_z + \bar{G}_z\tau \end{aligned} \tag{4.2}$$

where $\mathbf{\Phi} \in \mathbb{R}^2$ is a vector related to roll and pitch angle which are state in fully actuated subsystem, and $\tau \in \mathbb{R}^4$ is the control input of the multirotor which is a total thrust and torques. Other matrices $\bar{G}_x$, $\bar{F}_z$, and $\bar{G}_z$ are assigned to be equivalent to the dynamics of the nominal multirotor.

For comparison, the dynamics with external forces and torques exerted to the multirotor are written as follows:

$$\begin{aligned} \ddot{\mathbf{x}}_c &= G_x\mathbf{\Phi} + \mathbf{\Delta}_x \\ \ddot{\mathbf{z}} &= F_z + G_z\tau + \mathbf{\Delta}_z \end{aligned} \tag{4.3}$$

where $\mathbf{\Delta}_x$ and $\mathbf{\Delta}_z$ are disturbances owing to external forces and torques. Assuming that multirotor's attitude can follow its reference sufficiently fast, $\mathbf{\Phi}$ in the underactuated system dynamics

can be replaced with $\mathbf{\Phi}_r$ which functions as an input to the underactuated subsystem. Therefore, by generating a new reference for $\mathbf{\Phi}$ through applying extended high gain observer to the underactuated subsystem and by constructing DOB for the fully actuated subsystem, a nominal dynamics of the multirotor can be restored in the presence of external disturbances. The detailed algorithm for high gain observer and disturbance observer and related stability proof can be found in [6].

4.3 Hardware setup

We use a DJI F550 multirotor frame with 2312E motors controlled by 420Lite electronic speed controllers. The robotic arm is composed of ROBOTIS dynamixel XH430 series, and frames of OPEN MANIPULATOR-X. Joints in the robotic arm are velocity-controlled by a built-in controller provided by ROBOTIS. As the onboard computer, Intel NUC running Robot Operating System (ROS) in Ubuntu 16.04 executes MPC-based trajectory planner, DOB-based robust position controller, and navigation algorithm with VICON. The vehicle's attitude is controlled by the onboard low-level controller in Pixhawk 2. In addition, to address a realistic scenario, a custom-built door with its width and height of $D_w = 1.2$ m, and $D_h = 1.6$ m is employed. The weight of the door is about 11 kg, and with these values, the mass moment of inertia of the door can be calculated as $I_D \approx 5.28$ kgm^2.

5 Results

The proposed algorithms are validated with simulations and real experiments. This chapter presents both simulation and experimental results along with discussions.

5.1 Simulation results

Simulations were conducted to evaluate whether constraints are satisfied throughout the task. We validate that our algorithm could successfully generate a collision-free trajectory to various situations with different physical parameters of the door. Four different set of parameters were tested: $(D_D$ [Ns/rad], K_D [N/rad]$) = (0, 0), (10, 10), (50, 10), (10, 50)$. Simplified dynamics derived in the equation (2.9) is used for forward dynamics simulation while control input from MPC is directly applied to the forward dynamics simulator.

As can be seen in the figure 5.3, all values related to constraints are above the red line which signifies the constraint violation bound. Therefore, the concatenated constraints $c(\mathbf{x}_k) \geqq 0$ are all satisfied during whole operation. Furthermore, since the gray shaded regions in the figure 5.3 which denote constraint related values during predicted trajectories are all above the red line, constraints are all guaranteed during the predicted trajectories. Time history of the system state can be found in the figures 5.4 and 5.2 while 3 dimensional illustration can be found in the figure 5.1. Parameters in MPC are listed in the table 5.1.

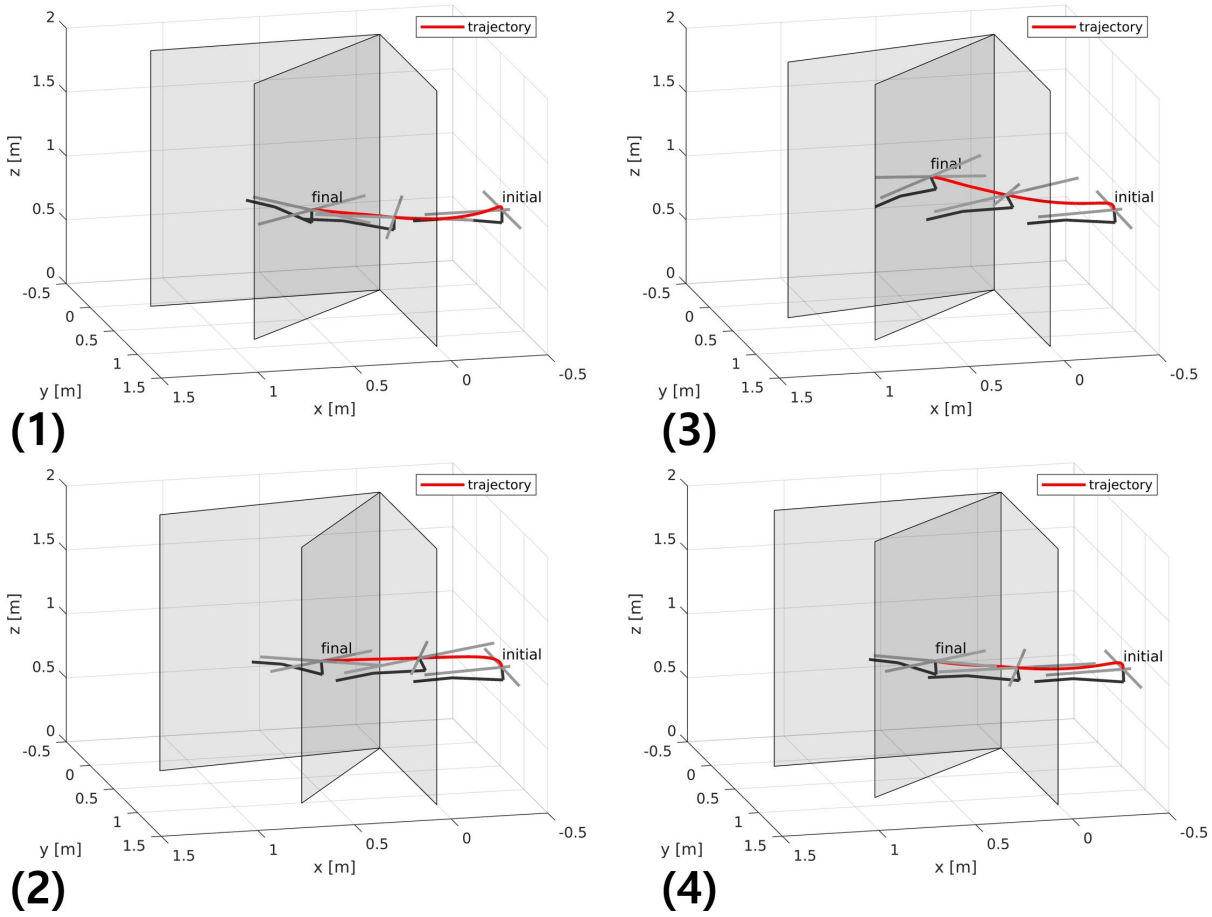

Figure 5.1: 3 dimensional visualization of the simulation results: (1) $(D_D, K_D) = (0, 0)$, (2) $(D_D, K_D) = (50, 10)$, (3) $(D_D, K_D) = (10, 50)$, (4) $(D_D, K_D) = (10, 10)$. The red curve is the computed trajectory, and the intermediate door illustration in all four scenarios depicts the pose of the hinged door at 1.2 seconds.

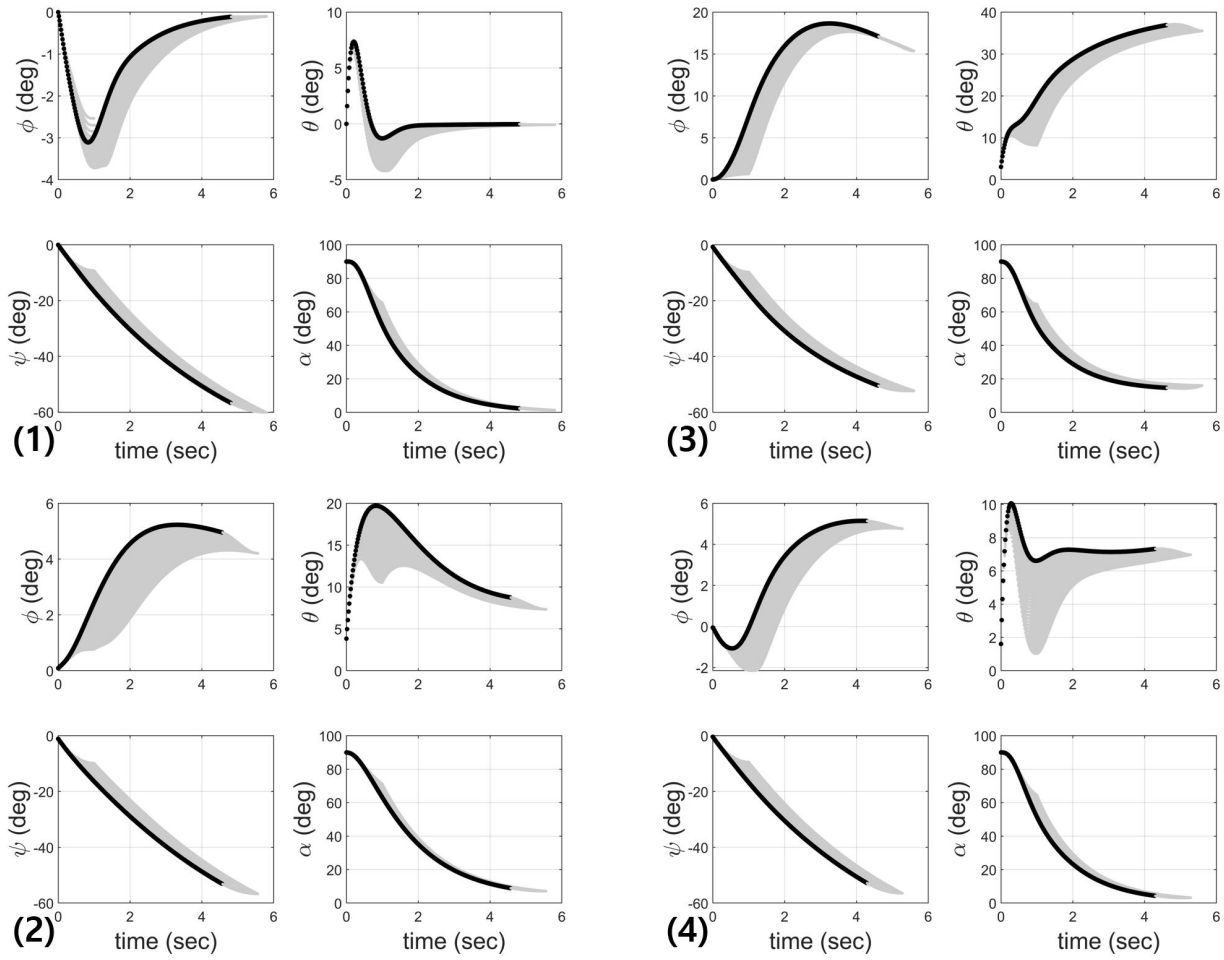

Figure 5.2: time history of ϕ, θ, ψ, and α: (1) $(D_D, K_D) = (0, 0)$, (2) $(D_D, K_D) = (50, 10)$, (3) $(D_D, K_D) = (10, 50)$, (4) $(D_D, K_D) = (10, 10)$. Shaded areas indicate accumulated predicted state trajectories. High spring constant of the door results in high pitch angle as in 3).

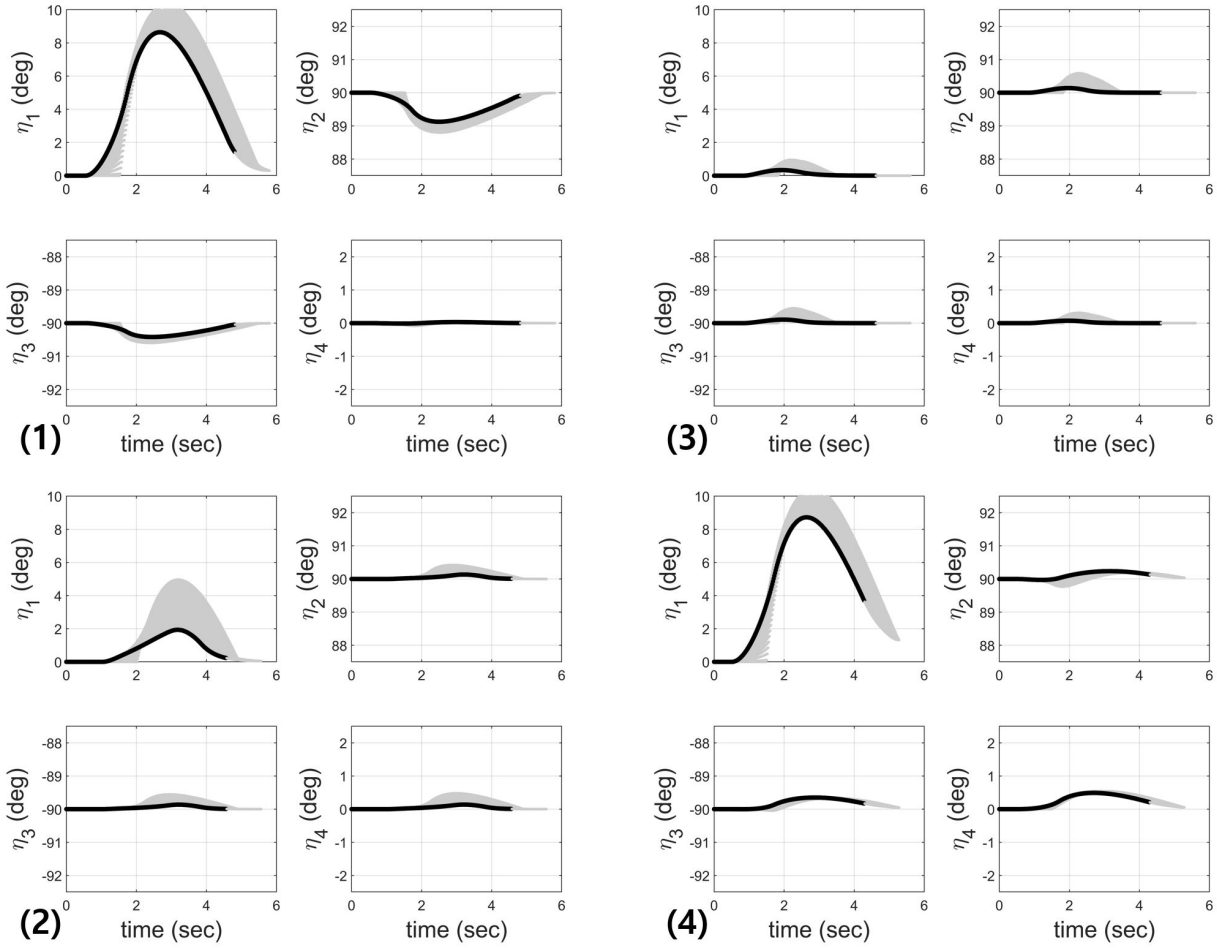

Figure 5.3: time history of η_1, η_2, η_3, and η_4: (1) $(D_D, K_D) = (0, 0)$, (2) $(D_D, K_D) = (50, 10)$, (3) $(D_D, K_D) = (10, 50)$, (4) $(D_D, K_D) = (10, 10)$. The scenarios 1) and 3) at which the door has no or small damping and spring constants show bigger motion in η_1 which rotates in z_B axis. This can be interpreted with the fact that interaction with the door of smaller damping or spring constant likely results in faster motion of the hinged door, for which in consequence, faster yaw motion is required. However, since faster yaw motion would lead to collision with the doorframe, first joint angle is twisted to avoid collision. Shaded area indicates accumulated predicted state trajectories.

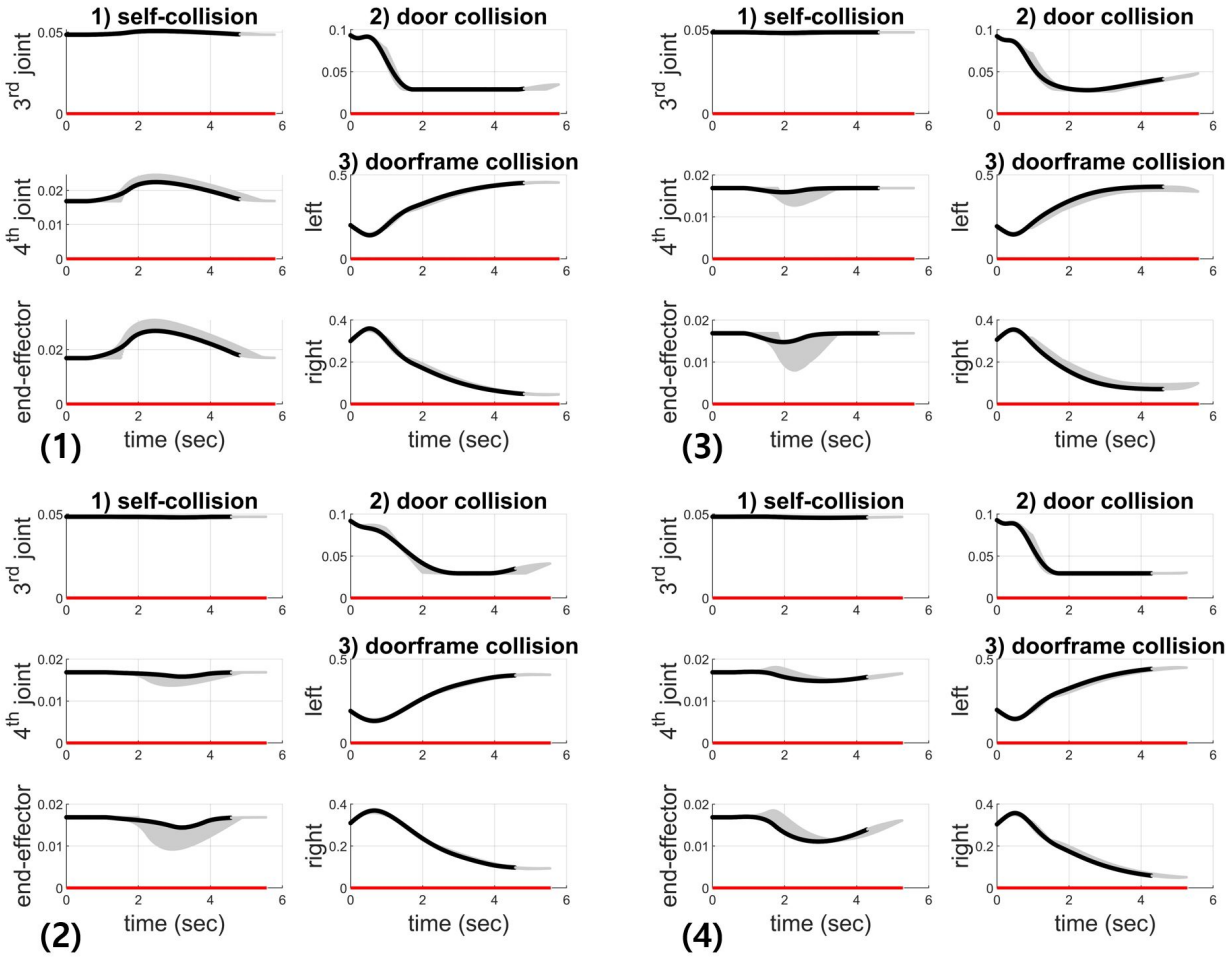

Figure 5.4: Constraint satisfaction check for each simulation: (1) $(D_D, K_D) = (0, 0)$, (2) $(D_D, K_D) = (50, 10)$, (3) $(D_D, K_D) = (10, 50)$, (4) $(D_D, K_D) = (10, 10)$. The red line denotes a constraint violation bound. If all values are above this red line as in the figure, all constraints are satisfied. Shaded areas show constraint violation check for every predicted trajectory, and one can find that all shaded regions are also above the red line.

Table 5.1: MPC parameters for both simulation and experiment

Parameters	Simulation	Experiment
L	$diag$[5 5 3 10 7 0.05 0.1 0.1 0.1]	$diag$[5 5 3 9 8 0.05 0.1 0.1 0.1]
Q	$diag$[5 5 3 10 7 0.05 0.1 0.1 0.1]	$diag$[5 5 3 9 8 0.05 0.1 0.1 0.1]
R	$diag$[0.1 5 5 13.5 0.01 0.01 0.01 0.01]	$diag$[0.1 5 5 13.5 10 10 10 10]
$\mathbf{x}_0$	$[0\ 0\ 0\ \frac{\pi}{2}\ 0\ 0\ \frac{\pi}{2}\ -\frac{\pi}{2}\ 0]^T$	$[0\ 0\ 0\ \frac{\pi}{2}\ 0\ 0\ \frac{\pi}{2}\ -\frac{\pi}{2}\ 0]^T$
$\mathbf{x}_i^d$	$[0\ 0\ -\frac{\pi}{2}\ 0\ 0\ 0\ \frac{\pi}{2}\ -\frac{\pi}{2}\ 0]^T$	$[0\ 0\ -\frac{7\pi}{18}\ \frac{\pi}{9}\ 0\ 0\ \frac{\pi}{2}\ -\frac{\pi}{2}\ 0]^T$

5.2 Experimental results

In the experiment, an aerial manipulator pushing a customized hinged door is conducted. Parameters of the door are chosen as $I_D = 5.28$ kgm^2, $D_D = 0$ N/rad and $K_D = 0$ Ns/rad while MPC parameters are listed in the table 5.1. The flow chart of the overall experimental scenario is presented in the figure 5.5. $\mathbf{\Phi}$, α, $\dot{\alpha}$, and $\mathbf{H}$ with f as their superscript denotes final target state $\mathbf{x}_s^f = \mathbf{x}_i^d$.

After the planner module receives all state information, it generates state and input trajectories. These trajectories are translated back into the aerial manipulation's desired position and velocity by the state converter, using a similar process as in the equation (4.1). The robotic arm in the flow chart contains an inherent velocity controller, and therefore desired velocity is depicted to be directly published to the robotic arm.

Although the input trajectory subscribed from MPC could be directly applied to the flight controller (FCU) for attitude control, which is Pixhawk 2 in our case, it is hard to guarantee stability during flight. Consequently, we adopt a disturbance observer(DOB)-based robust controller introduced in the section 4.2 for position control where its stability is fully analyzed in [6] along with experimental validations. This controller generates the desired attitude and angular velocities again to the flight controller by which we can ensure the aerial manipulation's stability.

Thanks to the capability of considering the state constraints in planning trajectory, the aerial

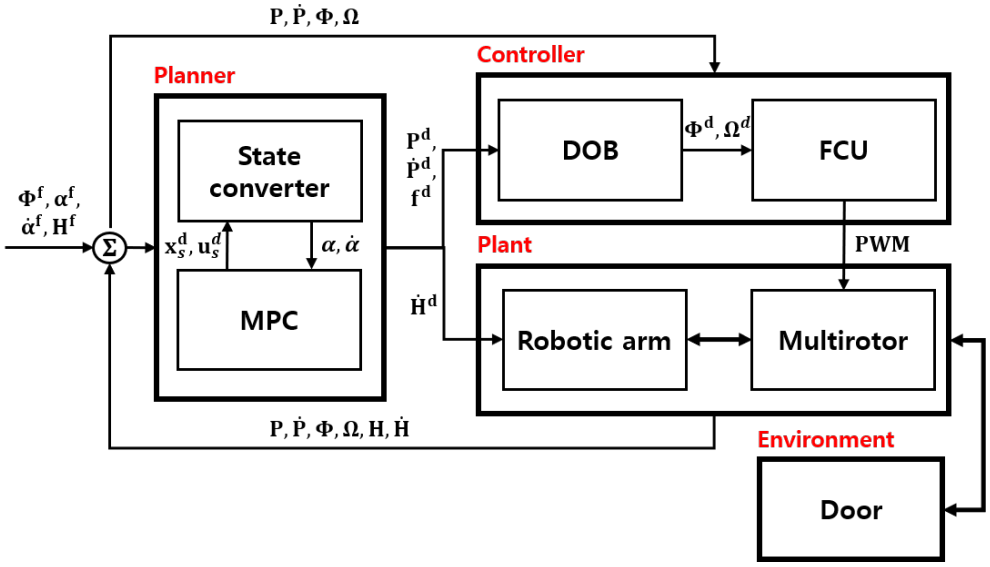

Figure 5.5: Overall control and planning structure of the aerial manipulation system.

manipulator successfully opens the door without collision as illustrated in Fig. 5.6. The history of the state during the experiment is described in Fig. 5.7. As expected, the door angle α tends to converge to the desired final value, implying that the door is sufficiently opened. Furthermore, followed by the changes in the door angle α, the vehicle's yaw motion ψ rotates accordingly. However, discrepancies between the desired and measured states, especially in α and ψ, occur due to the fact that the assumption 3 happens to be violated intermittently during the experiment. Although it is assumed that the end-effector is firmly attached to the door surface, uncertainty in door parameters and unmodeled dynamics between the UAM and the door seem to cause a faster door movement which results in a detachment between the door surface and the end-effector. Force control strategy like impedance control seems to be capable of handling this problem, and we leave it as a future work.

Figure 5.6: A composite image of an aerial manipulator opening a hinged door in the direction of the blue arrow. A transparent figure is the initial state of the system, and a vivid figure represents the final state of the system after successfully opening the hinged door.

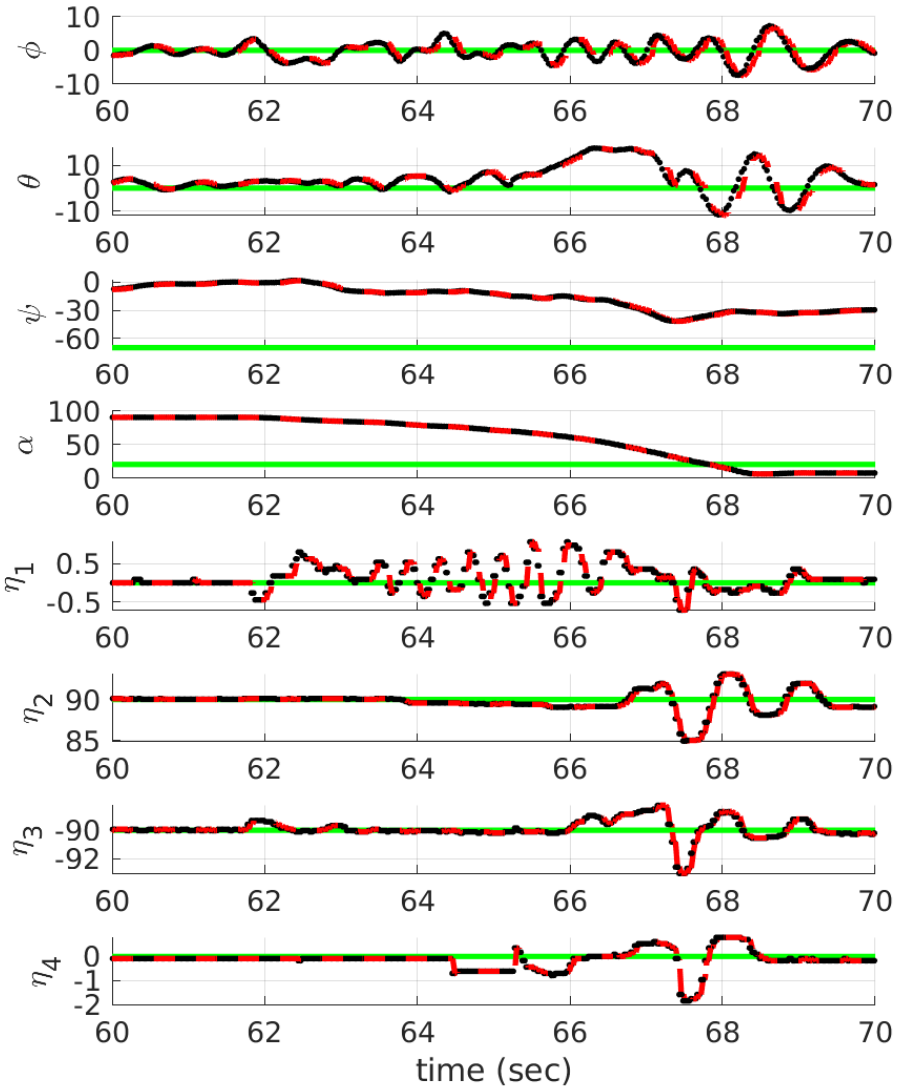

Figure 5.7: History of the states during the door opening experiment. The black line represents measured value. The dashed red line describes the predicted state from the MPC module. The green line represents the desired state of the system $\mathbf{x}_f$. All units on the y axis are degree.

6

Conclusion

In this paper, a systematic methodology for an aerial manipulator operating a hinged door is presented. Coupled equations of motion encompassing the aerial manipulator and the hinged door are first derived and later simplified to be applicable to an online-solvable optimal control problem. State constraints guaranteeing a safe trajectory are proposed, and the formulated MPC problem is solved with a constrained DDP algorithm. Generated trajectory is then tracked by a DOB-based robust controller, which provides stability during execution. The proposed algorithms are validated through simulations and experiments with a real-like door. Since this paper only presents preliminary experimental results, additional experiments are expected for examining repeatability and robustness of the proposed algorithm. Furthermore, to handle vibration in height direction which seemed to occur due to unexpected detachment between the end-effector and the door surface, designing a controller which can directly handle the detachment would be considered. For future studies, along with the one mentioned in the subsection 5.2, we anticipate to conduct multiple experimental scenairos by varying the door parameters. Also, a robotic arm with higher degrees of freedom is expected to provide better maneuverability during interaction.

References

[1] F. Ruggiero, V. Lippiello, and A. Ollero, "Aerial manipulation: A literature review," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 1957–1964, 2018.

[2] H. W. Wopereis, W. L. Van De Ridder, T. J. Lankhorst, L. Klooster, E. M. Bukai, D. Wuthier, G. Nikolakopoulos, S. Stramigioli, J. B. Engelen, and M. Fumagalli, "Multimodal aerial locomotion: An approach to active tool handling," *IEEE Robotics & Automation Magazine*, vol. 25, no. 4, pp. 57–65, 2018.

[3] S. Hamaza, I. Georgilas, M. J. Fernandez, P. J. Sanchez-Cuevas, T. Richardson, G. Heredia, and A. Ollero, "Sensor installation and retrieval operations using an unmanned aerial manipulator," *IEEE Robotics and Automation Letters*, 2019.

[4] K. Alexis, G. Darivianakis, M. Burri, and R. Siegwart, "Aerial robotic contact-based inspection: planning and control," *Autonomous Robots*, vol. 40, no. 4, pp. 631–655, 2016.

[5] K. Bodie, M. Brunner, M. Pantic, S. Walser, P. Pfändler, U. Angst, R. Siegwart, and J. Nieto, "An omnidirectional aerial manipulation platform for contact-based inspection," *arXiv preprint arXiv:1905.03502*, 2019.

[6] S. Kim, S. Choi, H. Kim, J. Shin, H. Shim, and H. J. Kim, "Robust control of an equipment-added multirotor using disturbance observer," *IEEE Transactions on Control Systems Technology*, vol. 26, no. 4, pp. 1524–1531, 2017.

[7] D. Lee, D. Jang, H. Seo, and H. J. Kim, "Model predictive control for an aerial manipulator opening a hinged door," *제어로봇시스템학회 국제학술대회 논문집*, pp. 986–991, 2019.

[8] D. Lee, H. Seo, D. Kim, and H. J. Kim, "Aerial manipulation using model predictive control for opening a hinged door," *arXiv preprint arXiv:2003.08256*, 2020.

[9] M. Tognon, E. Cataldi, H. A. T. Chavez, G. Antonelli, J. Cortés, and A. Franchi, "Control-aware motion planning for task-constrained aerial manipulation," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 2478–2484, 2018.

[10] M. Tognon, H. A. T. Chávez, E. Gasparin, Q. Sablé, D. Bicego, A. Mallet, M. Lany, G. Santi, B. Revaz, J. Cortés *et al.*, "A truly-redundant aerial manipulator system with application to push-and-slide inspection in industrial plants," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 1846–1851, 2019.

[11] M. Orsag, C. Korpela, S. Bogdan, and P. Oh, "Dexterous aerial robots—mobile manipulation using unmanned aerial systems," *IEEE Transactions on Robotics*, vol. 33, no. 6, pp. 1453–1466, 2017.

[12] S. Kim, H. Seo, and H. J. Kim, "Operating an unknown drawer using an aerial manipulator," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2015, pp. 5503–5508.

[13] H. Tsukagoshi, M. Watanabe, T. Hamada, D. Ashlih, and R. Iizuka, "Aerial manipulator with perching and door-opening capability," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2015, pp. 4663–4668.

[14] G. Darivianakis, K. Alexis, M. Burri, and R. Siegwart, "Hybrid predictive control for aerial robotic physical interaction towards inspection operations," in *2014 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2014, pp. 53–58.

[15] M. Neunert, C. De Crousaz, F. Furrer, M. Kamel, F. Farshidian, R. Siegwart, and J. Buchli, "Fast nonlinear model predictive control for unified trajectory optimization and tracking," in *2016 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2016, pp. 1398–1404.

[16] C. Y. Son, H. Seo, T. Kim, and H. J. Kim, "Model predictive control of a multi-rotor with a suspended load for avoiding obstacles," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 1–6.

[17] D. Jang, J. Yoo, C. Y. Son, H. J. Kim, and K. H. Johansson, "Networked operation of a uav using gaussian process-based delay compensation and model predictive control," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 9216–9222.

[18] D. Brescianini and R. D'Andrea, "Computationally efficient trajectory generation for fully actuated multirotor vehicles," *IEEE Transactions on Robotics*, vol. 34, no. 3, pp. 555–571, 2018.

[19] H. Seo, D. Lee, C. Y. Son, C. J. Tomlin, and H. J. Kim, "Robust trajectory planning for a multirotor against disturbance based on hamilton-jacobi reachability analysis," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2019, pp. 3150–3157.

[20] B. Plancher, Z. Manchester, and S. Kuindersma, "Constrained unscented dynamic programming," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2017, pp. 5674–5680.

[21] Y. Tassa, N. Mansard, and E. Todorov, "Control-limited differential dynamic programming," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2014, pp. 1168–1175.

국 문 초 록

비행 매니퓰레이터는 3차원 공간 속에 빠르게 위치할 수 있는 비행체의 장점과 외부와의 상호작용이 가능한 로봇팔의 장점이 결합된 비행체로, 최근 물건 집고 옮기기부터 물품 운송까지 다양한 임무를 수행하기 위해 활발하게 연구되어 왔다. 그러나, 온전히 비행 매니퓰레이터의 가능성을 활용하기 위해서는 움직일 수 있는 외부 구조와의 상호작용과 같이 더욱 복잡한 임무 또한 수행할 수 있어야 할 것이다. 여러 종류의 움직일 수 있는 구조물 중 본 논문에서는 일상 속에서 쉽게 마주칠 수 있는 경첩문을 여는 멀티로터 기반의 비행 매니퓰레이터에 대해 제시한다. 정적인 구조물과의 상호작용과는 달리 동적인 구조물과의 상호작용에 있어서 발생할 수 있는 1) 구조물의 제약된 움직임, 그리고 2) 움직이는 구조물과의 충돌 회피의 2가지 추가적인 문제에 대해 다루었다. 이러한 문제를 다루기 위해 모델 예측 제어 (MPC)를 적용하였으며, 시스템 동역학에 대한 제약조건 및 충돌 회피에 대한 제약조건을 부여하였다. 멀티로터 기반의 비행 매니퓰레이터와 경첩문의 결합 시스템에 대한 동역학을 유도하였으며, 이후 모델 예측 제어에서의 빠른 계산 속도를 위해 단순화되었다. 충돌 회피에 대한 제약 조건은 모두 상태 변수로 표현되었으며, 비행 매니퓰레이터의 멀티로터 프레임과 로봇팔 사이의 충돌 (자기 충돌), 문과의 충돌, 그리고 문틀과의 충돌을 고려하였다. 미분 기반의 동적 프로그래밍 기법 (differential dynamic programming)에 제약조건이 고려된 알고리즘을 구현함으로써 모델 기반 예측 제어를 통해 실시간으로 경로를 계획할 수 있다. 제안된 방법은 시뮬레이션과 실제 크기의 문을 활용한 실험을 통해 검증되었으며, 외란 관측기 기반의 강건 제어 기법이 실험에 활용되었다.

주요어 : 비행 매니퓰레이터, 모델 예측 제어, 충돌 회피.

학번 : 2018-22356